

**Claims:**

1. A method of creating an image, said image being formed by rendering at least a plurality of graphical objects to be composited according to a first hierarchical structure representing a compositing expression for said image, said first hierarchical structure including a plurality of nodes each representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, said method comprising the steps of:
  - storing a second hierarchical structure for at least one node of said first hierarchical structure, said second hierarchical structure indicating at least an unobscured region of an object associated with said node;
  - partitioning a space in which said object is defined into a plurality of regions;
  - overlaying said second hierarchical structure on said object such that said object is substantially encompassed within said second hierarchical structure;
  - traversing said overlayed second hierarchical structure to identify any of said plurality of regions which include at least a portion of said unobscured region; and
  - creating said image by rendering said identified regions.
2. The method according to claim 1, said method including the further step of traversing said first hierarchical structure to detect said node including said second hierarchical structure.
3. The method according to claim 1, wherein said second hierarchical structure is traversed for each of said plurality of regions.
4. The method according to claim 1, said method including the further step of producing a map for said plurality of regions, wherein said map at least indicates any region which includes at least a portion of said unobscured region.
5. The method according to claim 4, wherein said map includes a flag for each of said regions.
6. The method according to claim 4, wherein said map is produced using run-length encoding.

7. The method according to claim 4, wherein said map is traversed in a predetermined order to determine said identified regions.

5 8. The method according to claim 1, said method including the further step of converting said first hierarchical structure into a right leaning hierarchical structure.

9. The method according to claim 1, wherein said first hierarchical structure is a graphic object tree.

10

10. The method according to claim 1, wherein said second hierarchical structure is a quadtree.

15

11. A method of creating an image, said image being formed by rendering at least a plurality of graphical objects to be composited according to a first hierarchical structure representing a compositing expression for said image, said first hierarchical structure including a plurality of nodes each representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, said method comprising the steps of:

20

storing a second hierarchical structure for at least one node of said first hierarchical structure, said second hierarchical structure indicating at least an unobscured region of an object associated with said node;

traversing said overlaid first hierarchical structure to detect said node including said second hierarchical structure;

25

partitioning a space in which said object is defined into a plurality of regions upon detecting said node;

overlying said second hierarchical structure on said object such that said object is substantially encompassed within said second hierarchical structure;

30

traversing said second hierarchical structure to identify any of said plurality of regions which include at least a portion of said unobscured region; and  
creating said image by rendering said identified regions.

35

12. The method according to claim 11, wherein said second hierarchical structure is traversed for each of said plurality of regions.

13. The method according to claim 11, said method including the further step of producing a map for said plurality of regions, wherein said map at least indicates any region which includes at least a portion of said unobscured region.

14. The method according to claim 12, wherein said map includes a flag for each of said regions.

15. The method according to claim 12, wherein said map is produced using run-length encoding.

16. The method according to claim 12, wherein said map is traversed in a predetermined order to determine said identified regions.

17. The method according to claim 11, said method including the further step of converting said first hierarchical structure into a right leaning hierarchical structure.

18. The method according to claim 11, wherein said first hierarchical structure is a graphic object tree.

19. The method according to claim 11, wherein said second hierarchical structure is a quadtree.

20. An apparatus for creating an image, said image being formed by rendering at least a plurality of graphical objects to be composited according to a first hierarchical structure representing a compositing expression for said image, said first hierarchical structure including a plurality of nodes each representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, said apparatus comprising:

storage means for storing a second hierarchical structure for at least one node of said first hierarchical structure, said second hierarchical structure indicating at least an unobscured region of an object associated with said node;

partitioning means for partitioning a space in which said object is defined into a plurality of regions;

overlying means for overlaying said second hierarchical structure on said object such that said object is substantially encompassed within said second hierarchical structure;

traversing means for traversing said overlaid second hierarchical structure to  
5 identify any of said plurality of regions which include at least a portion of said unobscured region; and

image creating means for creating said image by rendering said identified regions.

10 21. The apparatus according to claim 20, wherein said traversing means further traverses said first hierarchical structure to detect said node including said second hierarchical structure.

22. The apparatus according to claim 20, wherein said second hierarchical structure  
15 is traversed for each of said plurality of regions.

23. The apparatus according to claim 20, further comprising map producing means for producing a map for said plurality of regions, wherein said map at least indicates any region which includes at least a portion of said unobscured region.  
20

24. The apparatus according to claim 20, wherein said first hierarchical structure is a graphic object tree.

25. The apparatus according to claim 20, wherein said second hierarchical structure  
25 is a quadtree.

26. An apparatus for creating an image, said image being formed by rendering at least a plurality of graphical objects to be composited according to a first hierarchical structure representing a compositing expression for said image, said first hierarchical  
30 structure including a plurality of nodes each representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, said apparatus comprising:

storage means for storing a second hierarchical structure for at least one node of said first hierarchical structure, said second hierarchical structure indicating at least an  
35 unobscured region of an object associated with said node;

first traversing means for traversing said overlayed first hierarchical structure to detect said node including said second hierarchical structure;

partitioning means for partitioning a space in which said object is defined into a plurality of regions upon detecting said node;

5           overlaying means for overlaying said second hierarchical structure on said object such that said object is substantially encompassed within said second hierarchical structure;

          second traversing means for traversing said second hierarchical structure to identify any of said plurality of regions which include at least a portion of said  
10       unobscured region; and

          image creating means for creating said image by rendering said identified regions.

27.       The apparatus according to claim 26, wherein said second hierarchical structure  
15       is traversed for each of said plurality of regions.

28.       The apparatus according to claim 26, further including map producing means for producing a map for said plurality of regions, wherein said map at least indicates any region which includes at least a portion of said unobscured region.

29.       The apparatus according to claim 26, wherein said first hierarchical structure is a  
20       graphic object tree.

30.       The apparatus according to claim 26, wherein said second hierarchical structure  
25       is a quadtree.

31.       A computer program for a computer comprising software code portions for performing a method of creating an image, said image being formed by rendering at least a plurality of graphical objects to be composited according to a first hierarchical structure  
30       representing a compositing expression for said image, said first hierarchical structure including a plurality of nodes each representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, said program comprising:

code for storing a second hierarchical structure for at least one node of said first hierarchical structure, said second hierarchical structure indicating at least an unobscured region of an object associated with said node;

code for partitioning a space in which said object is defined into a plurality of regions;

code for overlaying said second hierarchical structure on said object such that said object is substantially encompassed within said second hierarchical structure;

code for traversing said overlayed second hierarchical structure to identify any of said plurality of regions which include at least a portion of said unobscured region; and

code for creating said image by rendering said identified regions.

32. A computer readable medium storing a computer program, wherein said computer program comprises software code portions for performing a method of creating an image, said image being formed by rendering at least a plurality of graphical objects to be composited according to a first hierarchical structure representing a compositing expression for said image, said first hierarchical structure including a plurality of nodes each representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, said program comprising:

code for storing a second hierarchical structure for at least one node of said first hierarchical structure, said second hierarchical structure indicating at least an unobscured region of an object associated with said node;

code for partitioning a space in which said object is defined into a plurality of regions;

code for overlaying said second hierarchical structure on said object such that said object is substantially encompassed within said second hierarchical structure;

code for traversing said overlayed second hierarchical structure to identify any of said plurality of regions which include at least a portion of said unobscured region; and

code for creating said image by rendering said identified regions.

33. A method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the method comprising the steps of:

performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity for a node based on regions associated with the node; and

5 optimising said expression tree by performing a second traversal of said tree to determine obscurance information for each node of said tree using said opacity information, wherein said obscurance information represents at least an unobscured region associated with the node.

10 34. The method according to claim 33, wherein said opacity information is represented by a first hierarchical structure.

35. The method according to claim 33, wherein said obscurance information is represented by a second hierarchical structure.

15 36. The method according to claim 33, said method comprising the further step of identifying nodes having an associated complex graphical object.

20 37. The method according to claim 36, said method comprising the further step of determining opacity information for each node identified.

38. The method according to any one of claim 37, wherein said first hierarchical structure is dependent on said opacity information.

25 39. The method according to claim 33, wherein said first traversal is a bottom-up traversal.

40. The method according to claim 39, wherein opacity information of a child node is at least propagated to a parent node associated with said child node.

30 41. The method according to claim 33, wherein said second traversal is a top-down traversal.

35 42. The method according to claim 41, wherein obscurance information of a parent node is at least propagated to a child node associated with said parent node.

43. The method according to claim 34, wherein said first hierarchical structure is dependent on an operation associated with a node for which said first hierarchical structure is constructed.

5 44. The method according to claim 35, wherein said second hierarchical structures for a node are constructed by combining any first hierarchical structures associated with the node.

45. The method according to claim 34, wherein each leaf node of said first  
10 hierarchical structure is assigned a value depending on an opacity of a region associated with said leaf node.

46. The method according to claim 33, said method including the further step of converting said expression tree into a right leaning tree.

15 47. The method according to claim 34, wherein each node of said first hierarchical structure comprises a pointer indicating children nodes associated with said node.

48. The method according to claim 35, wherein said first and second hierarchical  
20 structures are quadtrees.

49. The method according to claim 33, wherein said opacity information is represented by bounding boxes.

25 50. The method according to claim 33, wherein said obscurance information is represented by bounding boxes.

51. A method for optimising an expression tree, said expression tree representing a  
compositing expression for compositing an image and comprising a plurality of nodes,  
30 each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the method comprising the steps of:



performing a first traversal of said tree to construct a first hierarchical structure for each operation node of said tree, wherein said first hierarchical structure represents combined opacity information for a node based on regions associated with the node; and

5       optimising said expression tree by performing a second traversal of said tree to construct a second hierarchical structure for each node of said tree using said opacity information, wherein said second hierarchical structure represents at least an unobscured region associated with the node.

10       52.       A method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, the method comprising the steps of:

15       performing a first traversal of said tree to identify nodes having an associated complex graphical object;

performing a second traversal of said tree to determine opacity information for each node identified in said first traversal;

20       constructing a first hierarchical structure for each node of said tree based on said opacity information, wherein said first hierarchical structure represents combined opacity information for a node based on regions associated with the node; and

optimising said expression tree by performing a third traversal of said tree to construct a second hierarchical structure for each node of said tree using at least one first hierarchical structure, wherein said second hierarchical structure represents at least an unobscured region associated with the node.

25

53.       The method according to claim 52, wherein said first traversal is a bottom-up traversal of said tree.

30       54.       The method according to claim 52, wherein each node having an associated complex graphical object is tagged.

55.       The method according to claim 52, wherein said second traversal is a top-down traversal of said tree.

56. The method according to claim 55, wherein said opacity information is propagated down said tree.

57. The method according to claim 52, wherein a first hierarchical structure of a child node is at least propagated to a parent node associated with said child node.

58. The method according to claim 52, wherein said third traversal is a top-down traversal.

59. The method according to claim 58, wherein a second hierarchical structure of a parent node is at least propagated to a child node associated with said parent node.

60. The method according to claim 52, wherein said first hierarchical structure is dependent on an operation associated with a node for which said first hierarchical structure is constructed.

61. The method according to claim 52, wherein said second hierarchical structures for a node are constructed by combining any first hierarchical structures associated with the node.

62. The method according to claim 52, wherein each leaf node of said first hierarchical structure is assigned a value depending on an opacity of a region associated with said leaf node.

63. The method according to claim 52, wherein each node of said first hierarchical structure comprises a pointer to indicate children nodes associated with said node.

64. The method according to claim 52, wherein said first and second hierarchical structures are quadtrees.

65. An apparatus for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the apparatus comprising:

means for performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity for a node based on regions associated with the node; and

5 means for optimising said expression tree by performing a second traversal of said tree to determine obscurity information for each node of said tree using said opacity information, wherein said obscurity information represents at least an unobscured region associated with the node.

66. The apparatus according to claim 65, wherein said opacity information is  
10 represented by a first hierarchical structure.

67. The apparatus according to claim 65, wherein said obscurity information is represented by a second hierarchical structure.

15 68. The apparatus according to claim 65, said apparatus further comprising means for identifying nodes having an associated complex graphical object.

69. The apparatus according to claim 68, said method comprising the further step of determining opacity information for each node identified.  
20

70. The apparatus according to claim 65, wherein said first and second hierarchical structures are quadtrees.

71. The apparatus according to claim 65, wherein said opacity information is  
25 represented by bounding boxes.

72. The apparatus according to claim 65, wherein said obscurity information is represented by bounding boxes.

30 73. An apparatus for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the apparatus comprising:

mean for performing a first traversal of said tree to construct a first hierarchical structure for each operation node of said tree, wherein said first hierarchical structure represents combined opacity information for a node based on regions associated with the node; and

- 5 means for optimising said expression tree by performing a second traversal of said tree to construct a second hierarchical structure for each node of said tree using said opacity information, wherein said second hierarchical structure represents at least an unobscured region associated with the node.

10 74. An apparatus for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression, the apparatus comprising:

- 15 means for performing a first traversal of said tree to identify nodes having an associated complex graphical object;

means for performing a second traversal of said tree to determine opacity information for each node identified in said first traversal;

- means for constructing a first hierarchical structure for each node of said tree based on said opacity information, wherein said first hierarchical structure represents combined opacity information for a node based on regions associated with the node; and  
20 means for optimising said expression tree by performing a third traversal of said tree to construct a second hierarchical structure for each node of said tree using at least one first hierarchical structure, wherein said second hierarchical structure represents at least an  
25 unobscured region associated with the node.

75. The apparatus according to claim 74, wherein said first traversal is a bottom-up traversal of said tree.

- 30 76. The apparatus according to claim 74, wherein each node having an associated complex graphical object is tagged.

77. The apparatus according to claim 74, wherein said second traversal is a top-down traversal of said tree.

78. A computer program for a computer comprising software code portions for performing a method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the program comprising:

code for performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity for a node based on regions associated with the node; and

code for optimising said expression tree by performing a second traversal of said tree to determine obscurance information for each node of said tree using said opacity information, wherein said obscurance information represents at least an unobscured region associated with the node.

79. A computer readable medium storing a computer program, wherein said computer program comprises software code portions for performing a method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the program comprising:

code for performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity for a node based on regions associated with the node; and

code for optimising said expression tree by performing a second traversal of said tree to determine obscurance information for each node of said tree using said opacity information, wherein said obscurance information represents at least an unobscured region associated with the node.

80. A method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said

node having a region of the image represented by said node, the method comprising the steps of:

performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node; and

optimising said expression tree by performing a second traversal of said tree to determine compositing information for at least one node of said tree, wherein said compositing information for a node is determined using opacity information associated with the node, and wherein said compositing information represents at least obscured regions, load regions and regions to be composited for an object associated with the node.

81. The method according to claim 80, wherein said compositing information is represented by a first hierarchical structure.

82. The method according to claim 81, further comprising the step of identifying nodes of said tree, for which a first hierarchical structure is required, depending on opacity information associated with the node.

83. The method according to claim 80, wherein said opacity information comprises a second hierarchical structure representing an opacity of a region associated with a node.

84. The method according to claim 80, wherein said opacity information comprises a bounding box representing an opacity of a region associated with a node.

85. The method according to claim 81, wherein said first hierarchical structure is dependent on said opacity information.

86. The method according to claim 81, wherein said first traversal is a bottom-up traversal.

87. The method according to claim 83, wherein opacity information of a child node is at least propagated to a parent node associated with said child node.

88. The method according to claim 87, wherein opacity information of said parent node is determined by merging at least two second hierarchical structures.

89. The method according to claim 87, wherein opacity information of said parent node is determined by merging at least one second hierarchical structure and a bounding box.

5 90. The method according to claim 80, wherein said second traversal is a top-down traversal.

91. The method according to claim 88, wherein a first hierarchical structure of a parent node is at least propagated to a child node associated with said parent node.

10

92. The method according to claim 83, wherein said second hierarchical structure is dependent on an operation associated with a node for which said first hierarchical structure is constructed.

15 93. The method according to claim 83, wherein each leaf node of said second hierarchical structure is assigned a value depending on an opacity of a region associated with said leaf node.

20 94. The method according to claim 83, wherein each node of said second hierarchical structure comprises a pointer to indicate child nodes associated with said node.

95. The method according to claim 83, wherein said first and second hierarchical structures are quadtrees.

25

96. A method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the method comprising the steps of:

30

performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node; and

optimising said expression tree by performing a second traversal of said tree to construct a first hierarchical structure for at least one node of said tree, wherein said first hierarchical structure is constructed for a node using opacity information associated with the node, and wherein said first hierarchical structure represents at least obscured regions, load regions and regions to be composited, for an object associated with the node..

97. A method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the method comprising the steps of:

performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node;

identifying nodes of said tree, for which a first hierarchical structure is required, depending on opacity information associated with the node; and

optimising said expression tree by performing a second traversal of said tree to determine compositing information for each node of said tree identified in said first



traversal, wherein said compositing information is determined for a node using opacity information associated with the node, and wherein said compositing information represents at least obscured regions, load regions and regions to be composited for an object associated with the node.

5

98. The method according to claim 97, wherein said compositing information is represented by a first hierarchical structure.

99. The method according to claim 97, wherein said opacity information comprises a  
10 second hierarchical structure representing an opacity of a region associated with a node.

100. The method according to claim 98, wherein said opacity information comprises a bounding box representing an opacity of a region associated with a node.

101. The method according to claim 98, wherein said first hierarchical structure is  
15 dependent on said opacity information.

102. The method according to claim 97, wherein said first traversal is a bottom-up  
traversal.

20

103. The method according to claim 99, wherein opacity information of a child node  
is at least propagated to a parent node associated with said child node.

104. The method according to claim 103, wherein opacity information of said parent  
25 node is determined by merging at least two second hierarchical structures.

105. The method according to claim 103, wherein opacity information of said parent  
node is determined by merging at least one second hierarchical structure and a bounding  
box.

30

106. The method according to claim 98, wherein said second traversal is a top-down traversal.

107. The method according to claim 106, wherein a first hierarchical structure of a parent node is at least propagated to a child node associated with said parent node.

108. An apparatus for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the apparatus comprising:

means for performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node; and

means for optimising said expression tree by performing a second traversal of said tree to determine compositing information for at least one node of said tree, wherein said compositing information for a node is determined using opacity information associated with the node, and wherein said compositing information represents at least obscured regions, load regions and regions to be composited for an object associated with the node.

109. The apparatus according to claim 108, wherein said compositing information is represented by a first hierarchical structure.

110. The apparatus according to claim 109, further comprising means for identifying nodes of said tree, for which a first hierarchical structure is required, depending on opacity information associated with the node.

111. The apparatus according to claim 108, wherein said opacity information comprises a second hierarchical structure representing an opacity of a region associated with a node.

112. The apparatus according to claim 108, wherein said opacity information comprises a bounding box representing an opacity of a region associated with a node.

113. An apparatus for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the apparatus comprising:

means for performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node; and

means for optimising said expression tree by performing a second traversal of said tree to construct a first hierarchical structure for at least one node of said tree, wherein said first hierarchical structure is constructed for a node using opacity information associated with the node, and wherein said first hierarchical structure represents at least obscured regions, load regions and regions to be composited, for an object associated with the node..

15

114. An apparatus for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the apparatus comprising:

means for performing a first traversal of said tree to determine opacity information for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node;

means for identifying nodes of said tree, for which a first hierarchical structure is required, depending on opacity information associated with the node; and

means for optimising said expression tree by performing a second traversal of said tree to determine compositing information for each node of said tree identified in said first traversal, wherein said compositing information is determined for a node using opacity information associated with the node, and wherein said compositing information represents at least obscured regions, load regions and regions to be composited for an object associated with the node.

115. A computer program for a computer comprising software code portions for performing a method for optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes,

each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the program comprising:

code for performing a first traversal of said tree to determine opacity information  
5 for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node; and

code for optimising said expression tree by performing a second traversal of said tree to determine compositing information for at least one node of said tree, wherein said compositing information for a node is determined using opacity information associated  
10 with the node, and wherein said compositing information represents at least obscured regions, load regions and regions to be composited for an object associated with the node.

116. A computer readable medium storing a computer program, wherein said computer program comprises software code portions for performing a method for  
15 optimising an expression tree, said expression tree representing a compositing expression for compositing an image and comprising a plurality of nodes, each said node of said tree representing at least a component of said image or an operation for combining sub-expressions of said compositing expression and each said node having a region of the image represented by said node, the program comprising:

code for performing a first traversal of said tree to determine opacity information  
20 for each node of said tree, wherein said opacity information represents combined opacity information for a node based on regions associated with the node; and

code for optimising said expression tree by performing a second traversal of said tree to determine compositing information for at least one node of said tree, wherein said  
25 compositing information for a node is determined using opacity information associated with the node, and wherein said compositing information represents at least obscured regions, load regions and regions to be composited for an object associated with the node.